

* More functions for the VZ200, March '84: There is an error in the second column, just above the listing of the short BASIC program. It should be ... (Can be done directly by POKE 30945,175.)

ETI April 1985 — 117

More functions for the VZ200

HERE is a simple way to add *automatic line numbering and trace* functions to VZ200 BASIC. Automatic line numbering should be self-explanatory. However, the trace function may need some explanation. When attempting to debug a BASIC program, it is sometimes useful to see exactly what sequence of instructions the computer is interpreting. This is the function of the trace command. It prints out on the video the sequence of line numbers the computer (the interpreter) is stepping through when executing a program. This allows you to make sure the program is doing what you intended it to do. (Especially useful in the case of conditional GOTO's or GOSUB's).

As adding the trace functions (TRON and TROFF) is the simplest task, I will deal with that first.

Before running your program, type in POKE 31003,175 from the immediate mode (no line numbers *That's it!* This is equivalent to typing in 'TRON'. Now when you run your program, each time a new line is selected to be interpreted (or the same line number repeated) it will be printed on the video. To disable this function just type POKE 31003,0 from the command level. This simulates using the 'TROFF' command.

A drawback with this method is that you might only want to debug a small section of the program and so have to contend with sorting out that small section from the rest of the displayed line numbers. This can be simply overcome by adding POKE 31003,175 into your program with a line number which places it in the program just before where you want to start the trace. Then add POKE 31003,0 with a line number which places it where you want the trace to stop.

Auto

Now to deal with the slightly more complex 'AUTO' function. This function, when enabled, saves you the trouble of typing sequential line numbers when entering a program. This very useful function will automatically display the next line number when you hit 'RETURN' at the end of a line of program.

To do this you need to supply the starting line number and the increment between lines. Next you need to set a flag which tells the BASIC that the 'AUTO' function is enabled. (This must be done last or you will go into the 'AUTO' mode before you have supplied the starting line and increment.)

The starting line number must be POKed into locations 30946 and 30947, and the line increment POKed into locations 30948 and 30949. These have to be in two-byte form with the least significant bit (LSB) going

This article details how you can simply add automatic line numbering and TRON and TROFF trace functions to the Dick Smith VZ200 colour computer.

Steve Olney

into the first location of each pair, and the most significant bit (MSB) going into the second.

For the line increment this is no problem as long as you keep the increment below 255. (Most increments would normally be less than 100). Just POKE 30948, 'increment' and then POKE 30949,0 where 'increment' is less than, or equal to 255. (I usually use 10 or 20 as the increment.)

Of course, the line number would most likely be above 255, so you must convert your starting line number into two bytes where:

LINE NO. = (MSB * 256) + LSB

and where we:

POKE 30946, LSB and POKE 30947, MSB

Example: For a starting line number of 2000

MSB = INT(2000/256) = 7

LSB = LINE NO. - (MSB*256) = 208

So we must:

POKE 30946, 208

POKE 30947, 7

For those not content with trouble of calculating this every time the 'AUTO' mode is entered, I have written a short program to do this as well as to enable the 'AUTO' function itself. (Can be done directly by POKE 30934,175.) Use line numbers which will put it well out of the way of any main program you are entering.

0 CLS

10 INPUT "STARTING LINE NO. ":S

20 INPUT "INCREMENT ":I

30 MS=INT(S/256):LS=S-MS*256

40 POKE 30946,LS:POKE 30947,MS

50 MI=INT(I/256):LI=I-MI*256

60 POKE 30948,LI:POKE 30949,MI

70 POKE 30945,175

80 END

For convenience, type this small program in starting from line number 0. This will enable quick access by just typing 'RUN' and then 'RETURN'. However to run your program, you will now need to type, 'RUN xxxx', where 'xxxx' is the first line number of your program.

To exit from the 'AUTO' mode, type 'CTRL' and 'BREAK' simultaneously exactly the same way you exit or interrupt a BASIC program. Incidentally, BASIC will automatically exit from the 'AUTO' mode when the new line number would have been greater than 65529. (The maximum line

number allowed in this BASIC.)

A useful feature of this 'AUTO' function is that, if you specify line numbers which include previously entered lines, then not only is the line number displayed but also the statements previously entered.

The cursor is conveniently positioned at the end of the line ready for any additions to that line. This can be used as a convenient editing feature. For example, let us suppose you have entered your program and now wish to go through and make corrections. Enter the first line number of the program to be corrected and the appropriate line increment for that program. You can now single step through your listing and make corrections as you wish! Unfortunately, there is no simple way of decrementing the line number. (other than manually POKE-ing in location 30946).

Why So Simple?

How was I able to add these two functions so easily? Well, on close scrutiny of the VZ200 BASIC in ROM, I discovered that it was fundamentally similar to Level II TRS-80 BASIC. By finding the equivalent control areas in RAM for the VZ200 BASIC, and by experimentation, I was able to get the functions working.

Apparently, the machine code for the execution of the 'AUTO', 'TRON' and 'TROFF' functions is still present in the VZ200 BASIC ROM, but the interpreter has been altered so as not to recognise the commands as valid in an input text string.

Why the machine code would be present in the BASIC ROM but not enabled is a bit strange. Perhaps some functions were dropped in order to implement all functions provided on the multi-function keys.

A word of warning! Like all situations where you are patching software (especially when written by someone else), beware of yet-undiscovered gremlins. I take no responsibility for any havoc wreaked by same!

A more elegant and flexible approach would be to intercept the text interpreter and make it recognise the 'AUTO' and trace commands from the immediate command level, and perhaps add a line re-numbering command. But that's another story!

(216-7)